DOCUMENT RESUME

ED 365 519 SE 053 869

AUTHOR Hauge, Sharon K.

TITLE Functions and Relations: Some Applications from Database Management for the Teaching of Classroom Mathematics.

PUB DATE [93]

NOTE 16p.

PUB TYPE Guides - Classroom Use - Teaching Guides (For Teacher) (052)

EDRS PRICE MF01/PC01 Plus Postage.

DESCRIPTORS Computer Uses in Education; *Database Management Systems; Databases; *Functions (Mathematics); *Mathematical Applications; *Mathematical Models; *Mathematics Instruction; Secondary Education; Teaching Methods

IDENTIFIERS *Relations (Mathematics)

ABSTRACT

While functions and relations are important concepts in the teaching of mathematics, research suggests that many students lack an understanding and appreciation of these concepts. The present paper discusses an approach for teaching functions and relations that draws on the use of illustrations from database management. This approach has the advantage of allowing students to undertake projects which will increase their research skills and provides students with an opportunity to become involved in current, up-to-date technology. The paper focuses attention on the relational database model, which is based on ideas inherent in sets, relations, and functions. The paper shows how the abstract mathematical concepts of Cartesian product, relation, function, domain, range, union, intersection, difference, projection, and transitivity enter into this model. A number of concrete examples from database management that can be used in classroom teaching and which can easily be modified to fall within the individual student's repertoire of experience include: career information obtained from The Occupational Outlook Handbook, a beauty salon appointment schedule, missing persons files obtained from the FBI, blood bank information obtained from the Red Cross, and a database in a school counselor's office. (Author/AA)

Functions and Relations
Some Applications from Database Management
for the
Teaching of Classroom Mathematics

by

Sharon K. Hauge, Ph.D.

Professor of Mathematics
The University of the District of Columbia

Washington, D. C.

ABSTRACT

While functions and relations are important concepts in the teaching of mathematics, research suggests that many students lack an understanding and appreciation of these concepts. The present paper discusses an approach for teaching functions and relations that draws on the use of illustrations from database management. Interfacing computer science and technology with mathematics offers interesting possibilities for mathematics education. This approach has the advantage of allowing students to undertake projects which will increase their research skills and provides students with an opportunity to become involved in current, up-to-date technology.

The paper focuses attention on the relational database model, which is based on ideas inherent in sets, relations, and functions. We endeavor to show how the abstract mathematical concepts Cartesian product, relation, function, domain, range, union, intersection, difference, projection and transitivity enter into this model. In doing so, we suggest a number of concrete examples from database management which can be used in classroom teaching and can easily be modified to fall within the individual student's repertoire of experience. These examples include: career information obtained from "The Occupational Outlook Handbook," a beauty salon appointment schedule, missing person files obtained from the FBI, blood bank information obtained from The Red Cross, and a database in a school counselor's office.

Functions and relations are threads that run through many mathematics courses extending from junior high school through college. Unfortunately, many students lack both an understanding of the basic ideas inherent in these concepts as well as an appreciation of why these ideas are important. In a recent Mathematics Assessment of the National Assessment of Educational Progress, it was found that students lacked a clear understanding of functional concepts. For example, both seventh and eleventh graders experienced difficulty in evaluating the expression a + 7, for a = 3, when functional notation was used. They also experienced difficulty in determining the domains and ranges of functions. On one item eleventh graders were presented with a function machine which doubled each whole number that was dropped into the input slot. Students were asked to choose the best description of the output. Fewer than one-third of the algebra students could correctly identify the range (Swafford and Brown, 1988).

Difficulties posed by functional concepts do not stop at the pre-college level. Papers presented at a 1990 international research conference concerned with learning functions indicated that undergraduate students often have the erroneous notion that functional relationships indicate causality, a change in x produces a change in y. Moreover, such students often are unable to see and express functional relationships (Selden & Selden, 1990).

These results suggest that we need to search for improved ways for teaching functions and relations. Interfacing computer science and technology with mathematics offers interesting possibilities. Computers and calculators are currently being used for graphing functions and for "unpacking information" about functions from their graphs.

In this paper, we shall consider another computer-based approach to teach students to think more clearly about relations and functions and to compose them. This approach is suggested by illustrations from computer science which occur in the area of database management. Pointing out the linkage between databases and mathematics may not only help students in math classes better understand relations and functions, it can also lay useful groundwork for students who later take courses in computer science which include discrete structures and database concepts.

Database management has become a crucial technology in contemporary society for any organization which must obtain, store, and utilize large amounts of data. These data must be structured in such a way that the resulting database provides the organization quick access to essential information. It should also be possible to modify the database, both to delete data that are no longer needed, and to insert new data without destroying the integrity of the database.

There are three basic models for constructing databases--the network model which is based on sets, the hierarchical model which employs trees, and the relational model which is based on ideas inherent in relational algebra--sets, relations, and functions. We will focus our attention here on the relational model, showing how the abstract mathematical concepts Cartesian product, relation, function, domain, range, union, intersection, difference, projection and transitivity enter into this model. In doing so, we will suggest some concrete examples from database management which may be used in the classroom and can be easily modified to fall within the individual student's repertoire of experiences.

In the relational model, which is generally considered to be the best of the three classical models, the data are conceptually stored in tables of rows and columns. No row is allowed to be repeated. The following table which is based on data in the "The Occupational Outlook Handbook" may be considered as an abridged example. Students may wish to use the library, conduct interviews, or write to government agencies to obtain data to construct their own examples or to expand this example and in class discussion point out mathematical concepts that can be illustrated from their tables.

TOMORROW'S JOBS
CAREER INFO

OCCUPATION	EXPECTED GROWTH IN JOB OPPORTUNITIES THROUGH THE YEAR 2000	TRAINING REQUIREMENTS	AVERAGE STARTING SALARY
Animal Caretaker	Average	On-the-job training	$9,000
Computer Programmer	Much faster than average	Specialized training	$16,000
Electrical Engineer	Much faster than average	Bachelor's Degree	$23,000
Hotel Manager	Slower than average	Bachelor's Degree	$30,000
Lawyer	Much faster than average	J.D. Degree	$34,000
Mathematician	Average	Bachelor's Degree	$27,500
Printing Press Operator	Slower than average	Apprenticeship	$34,000
Public Accountant	Faster than average	Bachelor's Degree	$23,000
Registered Nurse	Much faster than average	Bachelor's Degree	$23,000

In computer science, we might think of this table as a file; a row as a record, and a column as a field. In database management, the table is called a "relation"; a row is called a "tuple"; and a column is called an "attribute." The set of possible values an attribute may assume is called "the domain of the attribute." Thus rather than the tuples consisting of a number of slots into which anything may be placed, there is a concept of an underlying domain of possible values from which each attribute may be chosen, and each tuple corresponds to the selection of one value from each domain.

Both construction and manipulation of relations are based on mathematical theory. If we look at the table, we see a subset of the Cartesian product of occupations with expected growth in job opportunities through the year 2000 with training requirements with average starting salaries. In essence, we see a subset of a Cartesian product, which is indeed the definition of a "relation" in mathematics.

In the table, we also note that each occupation which appears appears in one and only one row. Since the occupation uniquely determines the row, the row is a function of the occupation. Thus, if we let O be the set of occupations, and let R be the set of rows in the table, we have a function f: O ->R whose domain is the set of all occupations in the table, and whose range consists of the rows in the table. For this reason, occupation is called a "key" of this relation. Expected growth in job opportunities through the year 2000 is not a key, because, an average growth rate is associated with more than one row. While looking at the table, students might be asked to consider such questions as "Does a training requirement determine a row?" or "Does an average starting salary determine a row?" Training requirements do not functionally determine the rows. Hence, training requirements are not a key of this relation. Similarly, average starting salaries do not functionally determine the rows, and therefore, cannot be a key for this relation.

If we wish to access the data in a particular row, the data may be accessed by a key, which in this case is occupation. As an illustration, we will show how the average starting salary for a mathematician in our career info relation might be accessed using commands from SQL, a data access language, which has become a standard in the database processing industry.

```
SELECT AVERAGE_STARTING_SALARY
  FROM CAREER_INFO
    WHERE OCCUPATION = "MATHEMATICIAN".
```

In our relation, the occupation functionally determines the expected growth in job opportunities through the year 2000, the training requirements, and the average starting salary. In database management, this may be expressed as

occupation --> expected growth in job opportunities
occupation --> training requirements
occupation --> average starting salary.

In mathematics, we might view the above three lines as three functions, whose domains are the set of occupations listed in the table. As in-class exercises, such questions as the following might be posed.

> Let O be the set of all occupations, and let G be the set of all growth-descriptions listed in the table. Then the table defines a function f: O -> G whose domain G contains all the occupations listed in the first column of the table. List all the items in the range of f.

The students might then be asked to construct and answer similar questions about the domains and ranges of other functions based on the table.

Domains and ranges are very important to systems analysts, computer programmers, and data entry personnel. When a database project is designed, both the data input and information outputs must be defined in terms of meaning, possible values, and format. For example, it must be known how many spaces to allow for an average starting salary, as well as whether the average starting salary is to be expressed as an integer or a real number with two decimal places.

In some relations, as in the career-info relation above, a key consists of one attribute; in other relations, a key may be composite, meaning that it consists of more than one attribute. Situations such as this give rise to examples of functions of several variables. The following example is suggested by an appointment schedule in a beauty salon.

APPOINTMENT SCHEDULE

DATE	TIME	EMPLOYEE NAME	CUSTOMER	SERVICE
12/1/91	11:00 a.m.	Robin Mitchell	Sara Wilson	Relaxer
12/1/91	11:00 a.m.	Sam Stevens	Bob Wright	Haircut-short
12/1/91	12:00 p.m.	Robin Mitchell	Nance Hulse	Haircut-short
12/1/91	12:00 p.m.	Cindy Simi	Sara Wilson	Nail wraps

In this relation, the row is a function of the three variables, date, time, and employee name. The date alone does not determine a row, since we see the same date in four rows. The time alone does not determine a row, since we see the same time in more than one row. Similarly, the employee name alone does not determine a row. But a given date together with a given time together with a given employee's name do determine a row. Hence, a composite key for this table is date, time, and employee, and these three variables can be used to access the data in a particular row. Thus, if we let D be the set of dates; T be the set of appointment times, E be the set of employee-names, and R be the set of rows in the table, we have an example of a function

$$f: D \times T \times E \to R.$$

of three variables, D, T, and E whose domain is the Cartesian product $D \times T \times E$ and whose range consists of the rows in the table.

In the appointment schedule table, other functions of several variables may be pointed out, as examples

date, time, employee-name ----> customer

and date, time, employee-name ----> service

Questions such as the following might be posed about these functions:

> If D is the set of appointment dates, T is the set of appointment times, E is the set of employee-names, and C is the set of customer's names, then the table defines a function f: D x T x E -> C. List the items in the domain of f. List the items in the range of f.

The students might then be asked to formulate and answer a similar question based on the appointment schedule.

Properties of relations follow from the mathematical properties upon which they are based. A relation is a subset of a Cartesian product of sets. Hence a relation is a set, and we can perform the standard set operations on relations to produce new relations. The binary operations union, intersection, and difference are often used to access information in relational databases.

For the results of union, intersection, and difference to make sense in database management, the relations on which these operations are performed must be "union compatible." This means that these relations must have the same number of attributes and each corresponding attribute must have the same domain.

Each of the operations union, intersection, and difference is defined as you might think. Union combines the tuples of one relation with those of another relation, eliminating duplicates. Intersection finds the tuples common to both relations. Difference finds the tuples that occur in the first relation but not in the second relation. In class discussion, functions suggested by the relations resulting from these operations as well as the domains and ranges of these functions might be explored.

My students have enjoyed constructing relations based on missing person file and unidentified person file statistics which we obtained from The National Crime Information Center and illustrating the binary operations union, intersection, and difference using their relations. For example, if JUVENILES is a relations showing data for missing juveniles (persons under the age of eighteen) and ENDANGERED is a relation showing data for missing persons who have previously reported having been threatened, the union of JUVENILES and ENDANGERED is the relation ENDANGERED. The intersection of JUVENILES and ENDANGERED is a relation which provides information about JUVENILES who had been threatened. The difference of ENDANGERED and JUVENILES is a relation which provides information about missing persons over the age of eightteen, who had reported being threatened.

In the above example, the relations JUVENILES AND ENDANGERED were union compatible. However, it is sometimes necessary to extract data across two or more relations that are not necessarily union compatible. If the key of one relation is an attribute of the other relations, the join operator may be used to extract these data. Join essentially takes the Cartesian product of the relations that are involved, applies selection to select the rows that are needed, and then applies projection to obtain the needed columns. The following is an abridged example suggested by the type of database used by the Red Cross in the collection and storage of blood.

In the relation below, BARCODE is a number used to identify a blood donation. It is represented by a series of black bars and spaces of varrious widths such as those that are seen on many consumer products. SSN is the donor's social security number, and DATE is the date on which the donor gave the blood. It should be noted that SSN, the key of the donor relation, is also an attribute of the blood bank relation.

BLOOD BANK RELATION

BARCODE	SSN	DATE
50000	512-46-7237	1/18/90
58790	634-90-5823	7/7/90

DONOR RELATION

SSN	NAME	PHONE	TYPE	RH
512-46-7237	John White	224-1870	A	+
634-90-5823	Ann Peak	244-8397	AB	-

An example of the ways this database might be used occurs when a blood bank administrator is faced with an emergency caused by an earthquake or flood. He or she would want to select from the pool of donors those who have specified types of blood and have not given blood during the past three months, which would make them potentially eligible for a new donation. In assignments or discussion, students might be asked to develop other hypothetical problems for the administrator, determine the data needed to solve these problems, and decide how these data might be accessed.

Suppose, for example, an administrator needs to find the dates on which the donors with type A blood last gave blood. To connect the blood bank and donor relations, the Cartesian product of these relations is taken to obtain the relation:

BARCODE	SSN	DATE	SSN	NAME	PHONE	TYPE	RH
50000	512-46-7237	1/18/90	512-46-7237	John White	224-1870	A	+
50000	512-46-7237	1/18/90	634-90-5023	Ann Peak	244-8397	AB	-
58990	634-90-5023	7/7/90	512-46-7237	John White	224-1870	A	+
58990	634-90-5023	7/7/90	634-90-5023	Ann Peak	244-8397	AB	-

Since a person can have only one social security number, rows with nonmatching SSN's are meaningless. To obtain a meaningful relation from the relation above, we select the rows which have SSN's that match. This gives Rows 1 and 4:

50000	512-46-7237	1/18/90	512-46-7237	John White	224-1870	A	+.
58990	634-90-5023	7/7/90	634-90-5023	Ann Peak	244-8397	AB	-.

The administrator was interested in finding the dates on which those persons with type A blood last donated blood. We now select those rows where the blood type is A. This gives the relation:

50000	512-46-7237	1/18/90	512- 46-7237	John White	224-1870	A	+.

Finally, we project this relation over DATE to obtain the column

1/18/90.

It may be noted that selection of rows is a function whose domain is a relation and whose range is the set of prescribed rows; whereas projection, which is similar to functions which project over coordinates in mathematics, is a function whose domain is a relation and whose range is the set of prescribed columns.

The relational model, as we have seen, is based on conceptually storing data in tables. A question often considered in database design is "What should go into a single table?" Updating the data in some designs can have undesirable consequences. The following example is suggested by a database in a student counselling office.

SID	COUNSELOR	COUNSELOR'S AREA OF RESPONSIBILITY
639-46-7872	Anagnoson	non-native Engish speaking students
729-51-3324	Beatty	career curricula
839-92-2271	Block	students undecided abour career choice

This relation has obvious problems. What happens if the student whose (SID) student identification is 639-46-7872 drops out of school, and we delete his record? If we are not careful in the design of our tables, we will also lose the fact that Dr. Anagnoson counsels non-native English speaking students. The same relation can be used to illustrate an insertion anomaly. Suppose we want to store the fact that a new counselor Robbins has come on board, and he assists students in need of psychological help. We cannot enter these data into the relation until we have the SID of a student who actually requests his services.

The mathematical concepts function and transitivity help us resolve these difficulties. The problem with the above design is that a type of transitivity involving SID, COUNSELOR, and COUNSELOR'S AREA OF RESPONSIBILITY is present in the table:

SID functionally determines the counselor's name.
The counselor's name functionally determines the counselor's responsibility.

SID also functionally determines that student's counselor's responsibility.

SID, the counselor's name, and the counselor's area of responsibility all appear in the same table. This type of transitivity often leads to both deletion and insertion anomalies. To resolve the problems, we eliminate the transitivity in the above table by splitting the relation into two relations, one relation containing the attributes SID and COUNSELOR; the other relation containing the attributes COUNSELOR and COUNSELOR'S AREA OF RESPONSIBILITY.

In this paper we have examined some of the mathematical concepts inherent in the relational database model, particularly focusing on ways this model can be used to illustrate applications of functions and relations. These ideas have their roots in the work of the mathematician E.F. Codd, who observed that data stored in files designed to obey certain constraints behaved as mathematical relations (Codd, 1970). Since then, concepts inherent in relational algebra, relations and functions have been used both in accessing data and in constructing some of today's most popular database management products such as SQL/DS, DB2, ORACLE, and R:BASE. The liaison between such popular software and mathematics can be a stimulus to classroom teaching. The kinds of projects that the students can undertake will help develop their research skills. Bringing computer science concepts into the classroom provides students an opportunity to become involved in current, up-to-date technology, and students know as well as we do, that knowledge about computer applications will increase their opportunities for finding good jobs and for performing well in them.

REFERENCES

Codd, Edgar F. "A Relational Model of Data for Large Shared Databanks." Communications of the Association for Computing Machinery, 13 (June, 1970): 377-381.

Kroenke, David M. and Kathleen A. Dolan. Database Processing, Third Edition. Chicago: Science Research Associates, Inc., 1988.

Selden, Annie, A. and John Selden, Jr. "Do Students Have a Sense of Function?" Undergraduate Mathematics Education Trends, 6 (1991): 6.

Swafford, Jane O. and Catherine A. Brown. "Variables and Relations" in Results from the Fourth Assessment of Educational Progress, edited by Mary M. Lindquist, pp. 55-63. Reston: National Council of Teachers of Mathematics, 1988.

1990-1991 Occupational Outlook Handbook. Washington, D.C: Bureau of Labor Statistics, 1990.